

operations performed at times **t1** through **t4** for a given hardware lane of Lane 0 through Lane 3 are the same as described earlier with some adjustments.

[0113] In the illustrated embodiment, at time **t2**, Lane 1 through Lane 3 determines the true intra-group dependency value to use for generating its decompressed word. As described earlier, in some embodiments, only packets with a same index progress in the pipeline together. In addition, it was described earlier that the true intra-group dependent value for a given packet is the youngest of the packets older than the given packet. Therefore, due to the assigning of the packets to the lanes and only allowing packets with a same index to progress together in the pipeline, after time **t1** when the index is known for each of the assigned packets, the command (Cmd) of the packet is used to select between the Payload (Cmd=Miss) and the decompressed word from the adjacent lane (Cmd=Hit). It is known between time **t1** and **t2**, the true decompressed word for Packet 0 in Lane 0 is the packet read from dictionary **510**, and the true decompressed word for Packet 1 in Lane 1 is the Payload of Lane 1, and the true decompressed word for Packet 2 in in Lane 2 is the Payload of Lane 3. Some multiplexing circuits are used and comparators for the Cmd value. However, the latency may still be appreciably small, and accordingly, no intra-group fixup operations are needed prior to storing decompressed words in the write buffer.

[0114] Referring now to FIG. **14**, a block diagram illustrating one embodiment of compression processing **1400** is shown. In the illustrated embodiment, parallel execution is performed for a hybrid compression algorithm again using the hardware lanes Lane 0 through Lane 3 and dictionary **510**. As shown, at time **t0**, each of Lane 0 through Lane 3 is loaded with a respective one of Word 0 through Word 3. The values of Word 0 through Word 3 are the same as the values used in the illustrated embodiment of FIG. **8** and FIG. **12**. The operations performed at times **t1** through **t5** for a given hardware lane of Lane 0 through Lane 3 are the same as described earlier except performed in a different order.

[0115] As described earlier, it can be known ahead of time that Lane 0 has the oldest word of Word 0 through Word 3 and Lane 3 has the youngest word of Word 0 through Word 3. At time **t2**, a single read request is sent from Lane 0 to dictionary **510**. In addition, at time **t2**, the value of the youngest word, which is Word 3, is stored for a later update of dictionary **510**. For example, the value C of Word 3 can be stored in a register to be used for writing entry 7 of dictionary **510** and for forwarding (bypassing) the value C to a younger second group with Word 4 through Word 7. At time **t4**, the determination for finding the true intra-group dependent value begins. In some embodiments, the processing steps at time **t4** lasts for two or more pipeline stages. The comparisons are done at time **t5** and the compressed packets are generated without performing intra-group fixup operations.

[0116] In various embodiments, program instructions of a software application may be used to implement the methods and/or mechanisms previously described. The program instructions may describe the behavior of hardware in a high-level programming language, such as C. Alternatively, a hardware design language (HDL) may be used, such as Verilog. The program instructions may be stored on a non-transitory computer readable storage medium. Numerous types of storage media are available. The storage medium may be accessible by a computer during use to

provide the program instructions and accompanying data to the computer for program execution. In some embodiments, a synthesis tool reads the program instructions in order to produce a netlist comprising a list of gates from a synthesis library.

[0117] It should be emphasized that the above-described embodiments are only non-limiting examples of implementations. Numerous variations and modifications will become apparent to those skilled in the art once the above disclosure is fully appreciated. It is intended that the following claims be interpreted to embrace all such variations and modifications.

1-20. (canceled)

21. An apparatus comprising:

a table comprising a plurality of entries; and

compression circuitry comprising a plurality of hardware lanes, wherein in response to receiving an indication of a compression instruction, the compression circuitry is configured to:

assign a first group of two or more input words to the plurality of hardware lanes;

responsive to determining at least a first input word and a second input word of the first group of two or more input words correspond to a same entry of the table, generate for the first input word and the second input word:

a single read request for the table; and

a single write request for the table; and

generate a compression packet for each of the first input word and the second input word.

22. The apparatus as recited in claim **21**, wherein the circuitry is further configured generate an index for each word assigned to a lane of the plurality of lanes.

23. The apparatus as recited in claim **22**, wherein to determine the first input word and the second input word of the first group of the two or more input words correspond to the same entry of the table, the circuitry is configured determine the first input word and the second input word have a same index.

24. The apparatus as recited in claim **21**, wherein the circuitry is configured to generate an index corresponding to a given input word based at least in part on a hash of the given input word.

25. The apparatus as recited in claim **21**, wherein the circuitry is configured to determine whether to update the table with any of the input words of the first group prior to determining dependencies between input words of the first group.

26. The apparatus as recited in claim **25**, wherein the circuitry is configured to:

assign a second group of input words from the plurality of input words to the first plurality of hardware lanes, wherein the second group is different from the first group wherein for each table entry to be accessed, the circuitry is further configured to:

determine a youngest input word of the first group; and

forward the youngest word to the second group prior to determining dependencies between input words of the first group.